UNITED STATES AIR FORCE

AIR UNIVERSITY

AIR FORCE INSTITUTE OF TECHNOLOGY

Wright-Patterson Air Force Base, Ohio

THE CORRELATION BETWEEN SUBJECTIVE AND OBJECTIVE MEASURES OF CODED SPEECH QUALITY AND INTELLIGIBILITY FOLLOWING NOISE CORRUPTION

THESIS

AFIT/GE/EE/81D-30

Jeffrey A. Kayser
Lt (j.g.) USCG

AFIT/GE/EE/81D-30

THE CORRELATION BETWEEN SUBJECTIVE AND OBJECTIVE MEASURES OF CODED SPEECH QUALITY AND INTELLIGIBILITY FOLLOWING NOISE CORRUPTION

THESIS

Presented to the Faculty of the School of Engineering

of the Air Force Institute of Technology

Air University

in Partial Fulfillment of the

Requirement for the Degree of

Master of Science in Electrical Engineering

by

Jeffrey A. Kayser, B.S.E.E

Lt (j.g.) USCG

Graduate Electrical Engineering

December 1981

Accession For	
NTIS GRA&I	X
DTIC TAB	☐
Unannounced	☐
Justification	
By	
Distribution/	
Availability Codes	
Dist	Avail and/or Special
A	

Approved for public release; distribution unlimited.

Preface

Finding an objective measure of speech intelligibility has long been a goal of the speech communications analyst. Several automated methods have been developed, but none have performed as an overall speech intelligibility measure for widespread application to speech communications systems. Subjective listener testing remains the most reliable method of measuring speech intelligibility.

With the appearance of linear predictive coding in communications theory and in the field of speech synthesis, it is believed that this method could be applied to speech intelligibility measurements. This study examines the use of linear predictive coding for objective intelligibility scoring and develops a metric for that purpose.

I am deeply indebted to Mrs. Alisa Workman of the Biological Acoustics Branch of the Air Force Aerospace Medical Research Laboratory for her hours of assistance in the preparation of the voice data tape and the subjective listener testing. I wish to thank Mr. Richard Mc Kinley for the use of the acoustics laboratory and its equipment. I also wish to thank Captain Larry Kizer, my advisor, for his guidance, assistance, and encouragement during this study; and thanks also to Dr. Matthew Kabrisky and Major Ken Castor for their guidance, assistance, and review of this report.

Jeffrey A. Kayser

Table of Contents

List of Figures

List of Tables

Abstract

A scoring metric of speech intelligibility based on linear predictive coding (LPC) was developed and evaluated. The data base used for evaluating the metric consisted of a list of 50 words from the Modified Rhyme Test. The list was transmitted over a LPC-10 Vocoder with no background noise. The list was scored subjectively for intelligibility by a trained listener panel. The subjective scores were used to judge the effectiveness of the objective metric.

The LPC scoring metric was calculated for the list of words and compared to the subjective scoring. The intelligibility score for the objective scoring metric was 82.99% with a standard deviation of 14.41%. The score for the subjective listener testing was 84.91% with a standard deviation of 7.47%. This shows a possible correlation between the objective LPC scoring metric and standard subjective listener scoring methods.

THE CORRELATION BETWEEN SUBJECTIVE AND OBJECTIVE MEASURES OF CODED SPEECH QUALITY AND INTELLIGIBILITY FOLLOWING NOISE CORRUPTION

I. Introduction

There exists a need within the military to measure the quality and intelligibility of speech at the output of a communication system. Many methods exist to measure parameters such as signal-to-noise ratio and channel noise; however, there are very few methods available to quickly and easily measure quality and intelligibility of the speech output. These qualities are highly subjective and current methods of measurement involve many manhours of testing and evaluation. A quicker and more efficient method is needed to calculate these measures. Situations where speech intelligibility and quality measurements are needed include the evaluation of similar voice communications equipments and the evaluation of a system's ability to withstand jamming or other corruption without loss of the transmitted message. Intelligible communications are vital to the military in all areas of operations. The purpose of this thesis is to determine an objective measure of speech that can be used as a metric of speech intelligibility.

Current Subjective Measures

There are presently two major methods of testing for intelligibility. These two methods are: subjective listener testing, and objective measurement techniques.

These two titles are very broad in their meaning and, to understand speech intelligibility, a more detailed knowledge is needed of these methods. In a report by Chambers (Ref 2), the entire area of subjective listener testing was reviewed. This section is a summary of the main points found in that report.

Subjective listener testing is a method in which a talker reads some form of a test over a communications system to a panel of listeners. The listeners respond to the talker; and by evaluating their responses, a determination is made as to how intelligible the communications system functions. The types of tests used by the talker vary depending on the parameter being tested. Five categories of these tests have evolved: articulation tests, intelligibility tests, speech comprehension tests, speech interference tests, and subjective appraisal tests.

Articulation tests are speech sound recognition tests. They use single speech sounds or phonemes which have no normal linguistic distributional properties and carry no meaning. They are not words but just sounds. These sounds are difficult to understand, and their primary use is in evaluating two or more relatively good communications systems. The best known test of this type is the Nonsense Syllable Test.

Intelligibility tests are speech perception tests. They are used to evaluate the ability of a communications system to correctly convey speech in the form of messages.

They consist of a series of words, short phrases, or sentences which the listeners hear and attempt to identify. Intelligibility is scored as the percentage correctly identified. There are presently twelve major tests of this type, two of which are the Fairbanks' Rhyme Test and the Modified Rhyme Test. Table I shows a listing of fifty words used in one test run of the Modified Rhyme Test. Column 1 would be the actnal word spoken by the talker, while the listeners would select from one of the six in that row.

Speech comprehension tests differ from intelligibility tests in that the listener must do more mental processing. More information is passed to the listener before a response is required. The listener must then use the accumulated information to make a decision. Because the human thought process has a limited input data rate, these tests are used to examine a communications system for degradation when additional mental tasks such as flying an aircraft are performed. Three tests in this area are the Message Rate Efficiency Test, the Single Answer Sentence Test, and the Repetition Rate Test. An example would be the Single Answer Test, in which a simple question requiring a one phrase response (such as, ''What is your altitude?'') is posed to the listener. The percentage of correct answers yields the percent sentence compehension or speech comprehension.

Speech interference tests are speech audibility tests. They are used to show the ability of a listener to correctly hear phonemes or speech sounds sent over a communications

TABLE I

Modified Rhyme Test

TOOK	COOK	HOOK	SHOOK	BOOK	LOOK
GUST	DUST	RUST	MUST	BUST	JUST
GANG	HANG	SANG	FANG	BANG	RANG
PEACH	PEAK	PEAL	PEAT	PEACE	PEAS
SUP	SUD	SUN	SUM	SUNG	SUB
BASS	BAD	BATH	BAT	BACK	BAN
PACK	PATH	PAD	PASS	PAN	PAT
PIN	PILL	PIP	PIT	PIG	PICK
COIL	BOIL	OIL	FOIL	SOIL	TOIL
SAD	SAG	SAT	SASS	SAP	SACK
DUG	DUD	DUB	DUCK	DUN	DUNG
TIP	RIP	DIP	LIP	HIP	SIP
CUFF	CUB	CUD	CUP	CUSS	CUT
GALE	MALE	SALE	TALE	PALE	BALE
DAY	SAY	WAY	GAY	MAY	PAY
LAW	RAW	PAW	SAW	THAW	JAW
TEST	BEST	NEST	VEST	WEST	REST
LAY	LACE	LAME	LAKE	LATE	LANE
FEAT	MEAT	HEAT	SEAT	NEAT	BEAT
BENT	TENT	RENT	DENT	SENT	WENT
BIG	DIG	PIG	WIG	RIG	FIG
SUN	RUN	GUN	NUN	FUN	BUN
HOT	LOT	NOT	POT	TOT	GOT
FIT	HIT	SIT	WIT	KIT	BIT
TEASE	TEACH	TEAL	TEAM	TEAK	TEAR
TACK	TAM	TANG	TAP	TAN	TAB
MAT	MAN	MAD	MASS	MAP	MATH
FIB	FIZZ	FIG	FIN	FILL	FIT
SHOP	COP	HOP	POP	TOP	MOP
WILL	TILL	KILL	FILL	HILL	BILL
SANE	SAKE	SAFE	SALE	SAME	SAVE
PANE	PACE	PAVE	PALE	PAGE	PAY
FEEL	KEEL	EEL	HEEL	REEL	PEEL
RED	SHED	BED	WED	FED	LED
KILL	KID	KIT	KINK	KIN	KICK
DIM	DIG	DIP	DID	DILL	DIN
SAME	CAME	GAME	TAME	NAME	FAME
PEN	HEN	MEN	THEN	DEN	TEN
CAVE	CAPE	CASE	CAME	CANE	CAKE
SIN	SING	SILL	SIT	SICK	SIP
PARK	BARK	LARK	DARK	HARK	MARK
PICK	TICK	KICK	LICK	SICK	WICK
DIN	PIN	SIN	TIN	FIN	WIN
BUCK	BUFF	BUT	BUS	BUG	BUN
FOLD	HOLD	GOLD	TOLD	SOLD	COLD
PUN	PUFF	PUP	PUCK	PUS	PUB
RAKE	RATE	RAVE	RAY	RAZE	RACE
BEAK	BEAT	BEAD	BEACH	BEAN	BEAM
SEED	SEEM	SEEN	SEEP	SEEK	SEETHE
HEAVE	HEAL	HEALTH	HEAP	HEAR	HEAT

system. These tests are more objective than other tests because they are based on signal-to-noise ratios. These signal-to-noise ratio curves are plotted for various frequencies over the audible range and these curves are used as performance curves of intelligibility. The most common interference test is the Articulation Index, or AI. This test is a very lengthy and detailed process, but it has become one of the more commonly used curve plotting methods. The Acoustical Society of America has written a complete standard on the proper method of calcuation of the Articulation Index (Ref 1). Although these tests are objective, they are classified as listener tests because they are not automatically calculated. They require plotting and evaluation by humans. This evaluation is a subjective evaluation.

The fifth group of tests are the Subjective Appraisal Tests. These tests require the listener to give an opinion of the communications system's performance. The listener judges the quality of the received speech. These tests are based on the confidence the listener has in what he heard and the effort required to understand the received speech. In the Confidence Ratings Test, the listener rates the confidence he has in what he heard. It uses a scale from 'positively received the message correctly' to 'positively received the message incorrectly.' These tests are used on low intelligibility systems.

Problem

When subjective testing methods are used to evaluate communications systems, they produce highly repeatable and useable results. The major drawback is that subjective testing is very expensive in both manpower and monetary costs. The talkers and listeners must be trained in the testing methods, and then they must undergo hundreds of hours of tests to obtain valid results. A typical testing panel consists of one talker, nine listeners, and one controller to run the test and evaluate the results. Special rooms are used to control background noise, and noise generators are required to add the necessary background noise found in the environment of the system being tested. All this is very costly when a system might be tested several times for just minor alterations.

A second drawback of subjective testing is that it cannot be easily performed in the actual system environment. Simulated conditions are used because such places as aircraft cockpits or battlefields prohibit on scene listener testing.

In 1969, the IEEE published the "Recommended Practice for Speech Quality Measurements." In this practice, the IEEE stated that since the start of the research for the paper, no generally applicable method of preference measurement had been developed (Ref 9:227). No standard method of intelligibility scoring has been developed that can be used as an overall guide to system performance. What is needed is a method to easily and quickly calculate an intelligibility score for a given system. It should be as close to real time operation as possible and require a minimum of personnel for proper operation. What is needed is a method to calculate an objecive measure of the system performance that can be used as a score of intelligibility. This method must be automated and correlate with subjective scores on the same system.

Current Objective Measures

Much research is being conducted to find an objective measure to replace subjective testing. Chambers' report (Ref 2) lists five automated methods in use in 1973. These earlier methods were the Pattern Correspondence Index, the Speech Communication Index Meter, the Voice Interference Analysis Set, the Automated Intelligibility Measurement, and the Sound Level Meter. Since 1973, other methods such as the Automatic Intelligibility Test Equipment and linear prediction have been developed.

The Pattern Correspondence Index measures the input and output of the system and integrates the difference between the two waveforms over the duration of the speech period. The integrated signal, presented as a meter reading, was related to the intelligibility of the system. The major problem was that the slightest delay in the system caused the two waveforms to be unmatched and the index failed.

The Speech Communication Index Meter (SCIM) is a device that automatically measures the Articulation Index (Ref 8:18). An earlier device called the Voice Interference

Analysis System (VIAS) also calculated the Articulation Index but did not perform in a reliable manner. The SCIM is a newer system modeled on the VIAS system. The VIAS system has very limited use while the SCIM can be extremely helpful in analysis of the time varying aspects of communications systems.

The Automated Intelligibility Measurement (AIM) uses computerized speech recognition as a technique to find word intelligibility scores. It is a method of phoneme matching in which a phoneme is sent over the system and the output is compared to a set of phoneme recognition 'masks' based on actual subjective testing. This method was developed into a system called Automatic Intelligibility Test Equipment (AITE) as a computer software package for the U. S. Air Force (Ref 10). This method is quite lengthy and time consuming. A complete set of masks is needed for each method and level of corruption used. The computer must compare the output phoneme with every mask and then determine the closest match. This system could never work in a real time environment.

The Sound Level Meter is a very simple tool that has been used to evaluate the impact of acoustic noise on speech communications. The major drawback of the meter is that it is an averaging device and the same meter readings may be obtained for a wide variety of spectrum shapes.

The newest area in automatic testing is Linear Prediction. "The basic idea behind linear predictive analysis is that a speech sample can be approximated as a linear combination of past speech samples. By minimizing the sum of the squared differences (over a finite interval) between the actual speech samples and the linearly predicted ones, a unique set of predictor coefficients can be determined'' (Ref 14:396). In a system being designed by Gamauf and Hartman (Ref 5), and later by Hartman and Boll (Ref 6), several different combinations of these predictor coefficients are being used to form an intelligibility score. No clear and exact method has been found to use these coefficients to produce a reliable and general method for scoring speech quality and intelligibility. However, there has been limited success in specific applications as is outlined in these reports.

General Approach and Assumptions

The purpose of this report is to describe a method of automatic intelligibility scoring using the objective measures of linear prediction. Linear prediction was selected because it is a new method and research has shown promise in using linear prediction for this purpose. It can be easily implemented on a computer and is geared toward the digital communications domain. Since the Department of Defense has decided to switch to digital voice communications, a scoring system or metric is needed that performs in the digital domain.

This study is very limited and will involve only the testing of one communications system, a real-time linear predictive vocoder currently under test by the U. S. Air Force. Since this vocoder is available, it will be used as the sample digital communications system under test. The assumption will be made that this vocoder is a good model of a digital communications system and could be of future use by the Department of Defense. Since linear predictive vocoder techniques are under intense study by all branches of the military, it is felt that this is a good assumption. This report will outline the theory of linear prediction, the procedures used to generate test data and subjective scores on the vocoder, and implementation of the linear predictive coding metric. The results of tests using the metric will be analyzed and correlated with the results of the subjective scores. Finally, conclusions will be drawn on these results and recommendations will be made as to the performance of the metric and on further study.

II. Linear Predictive Coding

Linear predictive coding (LPC) is a rapidly growing area of interest in the communications field. Various areas of application include voice encoders (or vocoders), speech recognition, and speaker identification. Although linear prediction has only had widespread use in the past fifteen years, it dates back to Gauss in 1795 (Ref 12:10) under the more descriptive title of linear least squares estimation. In 1975, John Makhoul published a tutorial review of linear prediction in the IEEE Proceedings (Ref 11). This chapter is based mainly on that review and presents the basic theory behind linear prediction and the autocorrelation solution algorithm of the linear prediction analysis model.

Model

The first step in the description of linear prediction is to discuss the model it is based upon. In applying discrete time series analysis to speech, each continuous-time signal $s(t)$ is sampled to obtain a discrete-time signal $s(nT)$ where n is an integer variable and T is the sampling interval. The sampling frequency is then $1/T$. Henceforth, $s(nT)$ shall be abbreviated by $s(n)$ or s_n with no loss in generality.

Consider a model in which the signal $s(n)$ is the output of a system with an unknown input $u(n)$ such that:

$$s(n) = -\sum_{k=1}^{p} a_k s(n-k) + G\sum_{l=0}^{q} b_l u(n-l), \quad b_0 = 1 \qquad (2.1)$$

where a_k, $1 \leq k \leq p$, b_l, $1 \leq l \leq q$, and G are the parameters of the hypothesized system and $s(n-k)$ are the past outputs. Equation 2.1 states that the output $s(n)$ is a linear combination of past outputs and past and present inputs, and thus is predictable from these past outputs and inputs. This results in the name linear prediction.

Equation 2.1 is in the time domain. By taking the Z transform of both sides and regrouping terms, the frequency domain model is given as:

$$H(z) = \frac{S(z)}{U(z)} = G\left[\left[1+\sum_{l=1}^{q} b_l z^{-l}\right] / \left[1+\sum_{k=1}^{p} a_k z^{-k}\right]\right] \qquad (2.2)$$

where

$$s(z) = \sum_{n=-\infty}^{\infty} s(n) z^{-n} \qquad (2.3)$$

is the Z transform of $s(n)$, and $U(z)$ is the Z transform of $u(n)$. Equation 2.2 is the generalized pole-zero model of the vocal tract. There are two special cases of this model that are of interest. The first case is the all-zero model where $a_k=0$, $1 \leq k \leq p$. This is known as the moving average model. The second case is the all-pole model where $b_l=0$, $1 \leq l \leq q$. This is referred to as the autoregressive model. The general pole-zero model is then called the autoregressive moving average model. The most widely used model for linear prediction of speech is the all-pole model, in which the numerator of Equation 2.2 is 1. This will be the model used in this chapter.

Linear Prediction of Speech

Linear prediction of speech is based upon the idea that a sample of a speech signal, s(n), can be approximated by a weighted sum of the preceding p samples of speech, where p is an integer. This yields the mathematical expression for s(n) as:

$$s(n) \simeq \sum_{i=1}^{p} a_i s(n-i) \tag{2.4}$$

where it is assumed that s(n) is the nth sample value of a speech signal, s(t), sampled every T seconds. Equation 2.4 is an approximation to the speech signal and, thus, is not exact. The error between the exact nth sample and its approximation can be defined as:

$$e(n) = s(n) - \sum_{i=1}^{p} a_i s(n-i) \tag{2.5}$$

The purpose of linear prediction is to find the weights (called predictor coefficients) that will minimize this error in some sense for a specified time interval.

The minimization technique selected is: the minimization of the total squared error over a specified time interval. This total squared error is defined as E, and is found by minimizing the expression:

$$E = \sum_{n}\Big[s(n) - \sum_{i=1}^{p} a_i s(n-i)\Big]^2 \qquad (2.6)$$

where the limits on n will define the interval over which the squared error is to be minimized. These limits will be discussed in detail later. To minimize Equation 2.6, the partial derivative is taken with respect to each predictor coefficient, a_i, and the result is set equal to zero. Doing this, the following result is obtained:

$$\sum_{n} 2\Big[s(n) - \sum_{k=1}^{p} a_k s(n-k)\Big]\Big[-s(n-i)\Big] = 0 \qquad (2.7)$$

where

$i=1,2,\ldots,p$

Rearranging terms and the order of summation in 2.7 gives the result:

$$\sum_{k=1}^{p} a_k \sum_{n} s(n-k)s(n-i) = -\sum_{n} s(n)s(n-i) \qquad (2.8)$$

Now the limits on n must be defined to solve Equation 2.8.

The selection of the solution technique specifies the limits on n. Two solution techniques outlined by Markel and

Gray (Ref 12) are the Covariance and the Autocorrelation Methods. In the Covariance Method, the minimization of E is defined for the interval of n=0, 1, ..., N-1 consecutive samples. The Autocorrelation Method defines the minimization of E for $-\infty < n < +\infty$. For this method, the speech signal is defined as:

$$s(n) = \begin{cases} s(n), & n = 0,1,\ldots,N-1 \\ 0, & \text{otherwise} \end{cases} \qquad (2.9)$$

This is done by using a window on the s(n) signal of a length N.

For this study, the Autocorrelation Method was selected because it insures a stable model (Ref 12:130) and requires fewer calculations in the solution.

<u>Autocorrelation Method</u>

Using the Autocorrelation Method, Equation 2.8 can be rewritten as:

$$\sum_{k=1}^{p} a_k \sum_{j=-\infty}^{\infty} s(n-k)s(n-i) = -\sum_{j=-\infty}^{\infty} s(n)s(n-i) \qquad (2.10)$$

Letting j=n-i yields:

$$\sum_{k=1}^{p} a_k \sum_{j=-\infty}^{\infty} s(j+i-k)s(j) = -\sum_{j=-\infty}^{\infty} s(j+i)s(j) \qquad (2.11)$$

and the estimate of the autocorrelation function of the signal s(n) is:

$$R(i) = \sum_{n=-\infty}^{\infty} s(n)s(n+i) \qquad (2.12)$$

where

$$R(i)=R(-i)$$

Using the previous definition of s(n) from Equation 2.9 yields:

$$R(i) = \sum_{n=0}^{N-1-i} s(n)s(n+i) \qquad (2.13)$$

when i=1, 2, ..., p, Equation 2.13 is defined as the short-term autocorrelation of s(n). Using this in Equation 2.11 yields:

$$\sum_{k=1}^{p} a_k R(i-k) = -R(i) \ , \ i = 1,2,...,p \qquad (2.14)$$

After the short-term autocorrelation is computed, Equation 2.14 represents p linear equations that can be solved simultaneously for each a_k. A recursive solution has been developed by Levinson that provides computational efficiency in solving these equations (Ref 12).

Levinson's Algorithm

Levinson's algorithm was selected by Markel and Gray for use in their Fortran IV subroutine AUTO (Ref 12:216), which provides a recursive solution to Equation 2.14. The following definitions are made to simplify notation:

$A_i^{(p)}$ = The ith predictor coefficient of the pth order model

r(n) = Normalized short-term autocorrelation coefficients

where

$$r(n) = \frac{R(n)}{R(0)} \quad (2.15)$$

With these definitions, Equation 2.14 can be written as:

$$-r(j) = \sum_{i=1}^{p} A_i^{(p)} r(i-j) \ , \ j = 0,1,\ldots,p \quad (2.16)$$

To start the algorithm, define a new quantity, K_0, as:

$$K_0^{(0)} = \frac{r(1)}{r(0)} \quad (2.17)$$

and recursively calculate $K_i^{(p)}$ using:

$$K_0^{(p)}\left[r(0)-\sum_{i=0}^{p-1} K_i^{(p-1)} r(p-i)\right] = r(p+1)-\sum_{i=1}^{p} K_{i-1} r(i) \quad (2.18)$$

and

$$K_i^{(p)} = K_{i-1}^{(p-1)} - K_0^{(p)} K_{p-i}^{(p-1)} \quad , \; i=1,2,\ldots,p \qquad (2.19)$$

Having calculated $k_i^{(p)}$ and defining $A_0^{(0)}=1$, $A_i^{(p+1)}$ can be calculated from:

$$A_{(p+1)}^{(p+1)}\left[r(0)-\sum_{i=0}^{p} K_j^{(p)} r(p+1-i)\right] = r(p-1)-\sum_{i=0}^{p} A_j^{(p)} r(p+1-i) \qquad (2.20)$$

and

$$A_i^{(p+1)} = A_i^{(p)} - K_i^{(p)} A_{(p+1)}^{(p+1)} \quad , \; i = 0,1,\ldots,p \qquad (2.21)$$

This method generates the two vector quantities $A^{(p)}$ and $K^{(p)}$. $A^{(p)}$ is a vector of the predictor coefficients for the filter model. $K^{(p)}$ is a vector of the reflection coefficients. Each $K_j^{(p)}$ is analagous to the reflection coefficients of a p-section transmission line. If any transmission line reflection coefficient is greater than 1, the circuit is unstable. This is also true for $K_j^{(p)}$; the Levinson algorithm generates all $K_j^{(p)}$ values less than 1 and therefore yields a stable model.

The third quantity generated by the algorithm is the minimum total squared error for the model. Setting $E_0=1$ and recursively solving for E_{p+1}, the following equation is generated:

$$E_{p+1} = E_p + A_{(p+1)}^{(p+1)}\left[R(p+1) - \sum_{i=0}^{p} K_i^{(p)} r(i)\right] \qquad (2.22)$$

This method of calculation of these three quantities is incorporated in the Fortran IV subroutine AUTO by Markel and Gray and is listed in the appendix of this report.

III. Procedure

In order to study Linear Prediction as a method of objectively determining intelligibility, a communications system was used that was also under testing by a subjective listener panel. The system being tested was the LPC-10 Vocoder developed at Lincoln Laboratory, Massachusetts Institute of Technology. This system was a microprocessor realization of a linear predictive vocoder and is described in detail in Reference 7. It used a tenth-order linear predictive approximation of the input signal, and was operated at 2400 bits per second. Two major points of using this system were that it was a real-time digital communications system and subjective listener testing was available.

Data Tape Recording

The first step in the anaysis process involved the recording of speech before and after it was processed by the vocoder. This recording was done at Wright-Patterson Air Force Base, in the facilities of the Biological Acoustics Branch of the Air Force Aerospace Medical Research Laboratory. This facility was also testing the vocoder using subjective listener tes ıg.

Figure 1 shows the general arrangement of equipment used to record the speech tape. The laboratory consisted of ten talker/listener desk modules inside a soundproof room that was equipped with the Air Force ESD 381 Jammer #1. This system is capable of generating, inside the room,

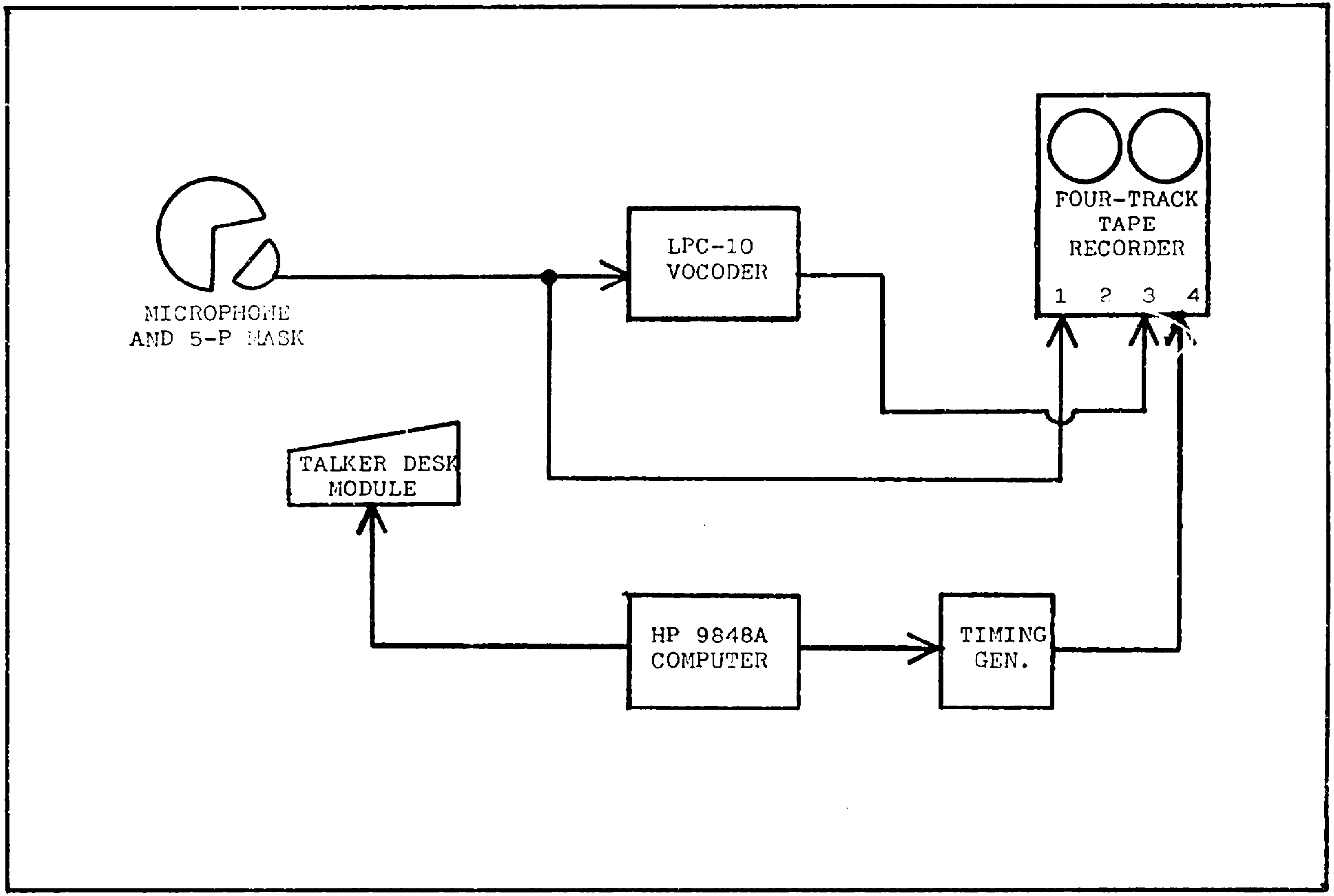

Figure 1. Data Tape Generation

cockpit noises that simulate various conditions for aircraft used by the Air Force. No background noise was used for the speech tape recording so that the no-noise operation of the vocoder could be tested.

A four track tape recorder was used to record both the input and output of the vocoder. A trained male talker sat in the room wearing a standard Air Force helmet with the 5P mask and M-101\A/C microphone. A Hewlett Packard 9848A computer was used to control the testing. It had 300 words from the Modified Rhyme Test stored in memory. This computer was also used to administer the Modified Rhyme Test for subjective listener testing.

One complete run of the Modified Rhyme Test was conducted to produce the speech tape. The HP 9848A computer created a display on the talker desk module instructing the talker to pronounce one of the words from the first column of Table 1. Every ten seconds, the computer changed the word until all fifty words were spoken. The complete test lasted approximately nine minutes for one fifty word list. The computer was also connected to a timing generator for tape alignment. At the start of each ten second word interval, a single timing mark was produced by the timing generator. At the end of each interval, two timing marks were generated. A timing mark consisted of a rapid transition from a zero level to a negative peak and then returned to a zero level. These timing marks were recorded directly onto track four of the speech tape for use in alignment of words in the analysis process to be described later.

When the talker saw a new word on his desk module, he spoke the number of the word, followed by a short phrase including the word. An example for the first word on the list would be: "Number 'one,' you will mark 'took,' please." A carrier phrase was used as in the Modified Rhyme Test because the talker pronounces the word differently if just the word is spoken (Ref 15). This is a major reason why the Air Force has selected the Modified Rhyme Test for use in subjective listener testing. Also, the carrier phrase is used by test equipment to maintain a constant level (using an automatic gain control) for recording purposes. The output of the talker's microphone was directly recorded on track one of the tape recorder. It was also connected to the input of the LPC-10 Vocoder. The output of the vocoder was recorded on track three of the tape recorder. Track two was used for any general comments by the administrator of the test. A tape of a complete fifty word test was produced. Track one contained the undistorted words directly from the speaker. Track three contained the distorted words from the output of the vocoder. Track four contained the timing marks. This tape was then taken to the Signal Processing Laboratory at the Air Force Institute of Techology for computer loading and analysis.

Computer Loading and Alignment of Speech Tape

The next step in the analysis process was to load the speech recorded on tape into the signal processing computer for analysis. Figure 2 shows the general arrangement of equipment necessary to load the speech data into the computer memory.

Each word was played back by the tape recorder and passed through a six-pole butterworth low-pass filter with a 3db cutoff frequency of 3.2 KHz and rolloff of 48 db per octave. Several reasons exist for using the 3.2 KHz cutoff point. The major point is that the analog-to-digital sampler operates at 8000 samples per second; and, in order to satisfy the Nyquist sampling theorem, the highest frequency component in the signal must be less than 4 KHz. A second point is that the first three formant frequencies of a male voice with an average vocal tract length of 17 cm will lie in the frequency range of about 250-2800 Hz (Ref 12:153). Shorter vocal tracts such as in women and children produce formats in the range of about 300-3500 Hz. This cutoff of 3200 Hz will allow the first three formants to pass and reject the higher formants. These first three formants are the major contributors to the speech waveform and are necessary for speech intelligibility (Ref 4:53). Still further, a third consideration is that the standard telephone system is band limited to a range of 300-3200 Hz, and from this limit, 3200 Hz was used so that the test system would conform to use over standard telephone communications systems.

The recorded words were played back, and after the

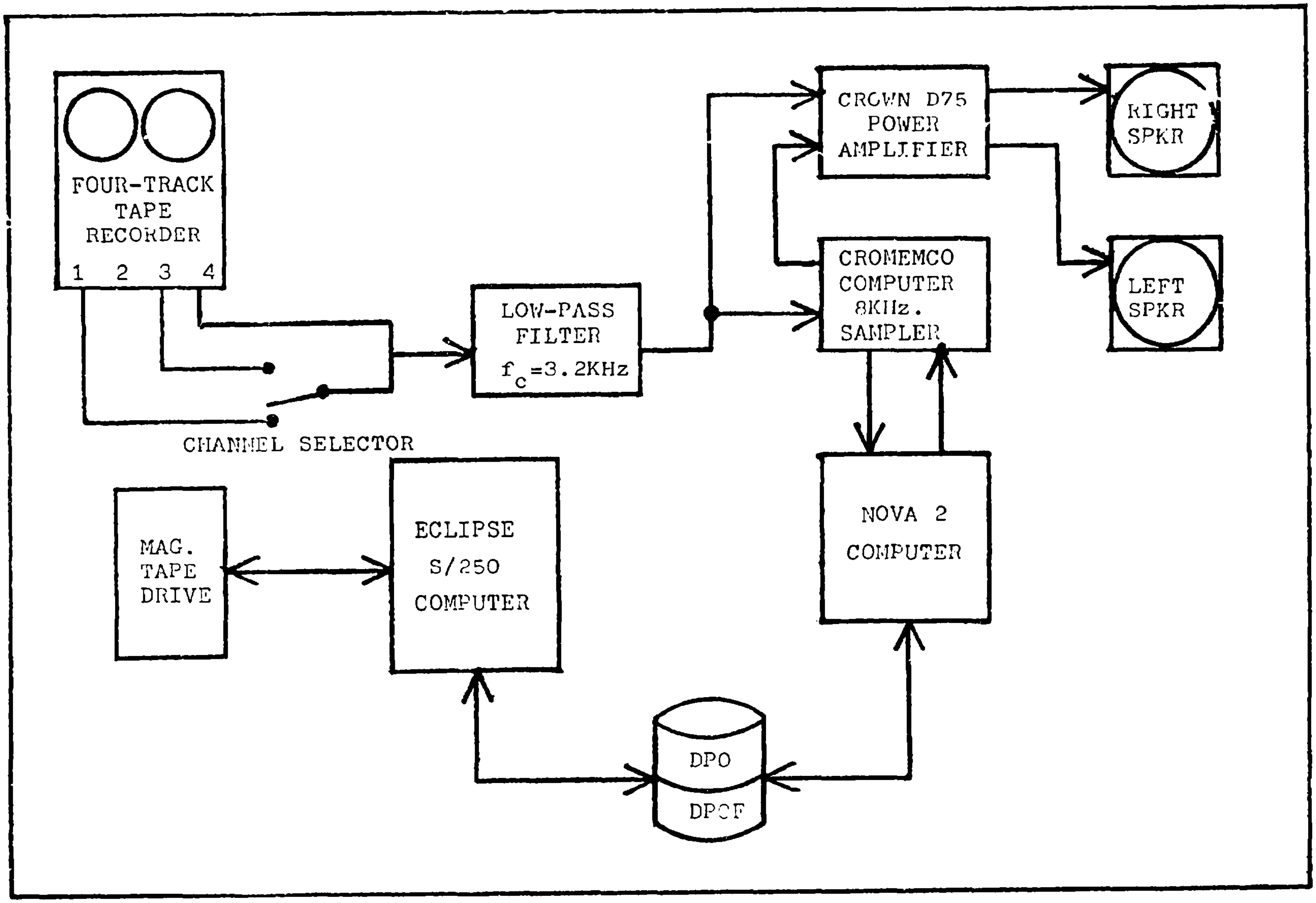

Figure 2. Computer Loading of Data Tape

filter, they passed through an audio amplifier to speakers. They also were input to the Cromemco Computer, which sampled the input at a rate of 8000 samples per second and stored the samples in memory in the form of 88 blocks of digital data, each block containing 256 samples. The net result was that 2.82 seconds of speech were stored for each word.

The Cromemco computer was controlled by the NOVA 2 computer. Capt. Paul Finkes, USAF, wrote for the NOVA several useful control programs that allowed data to be recorded via the Cromemco computer. The two main programs used were AUDIOHIST and AUDIOMOD. These programs are listed and described in Reference 3.

Actual analysis of the data was to be done on the ECLIPSE S/250 computer. Figure 2 shows that data files may be transferred from the NOVA to the ECLIPSE through a common disk directory DP0/DP0F. Since each word required 88 blocks of storage space and there were 100 words, 50 undistorted and 50 distorted, there was not enough space on disk to store all 100. Therefore, the word files were stored on magnetic tape from the ECLIPSE so that ample space would exist for processing and analysis programs.

The program AUDIOHIST was used to read in each word from tape. If Figure 2 is examined closely, it can be seen that as each word was played back, the timing marks on track four were also played back. The switch allowed selection between the undistorted words on track one and the distorted words on track three.

Once word pairs of undistorted and distorted words were read into the computer, they had to be aligned before any analysis could take place. This was the purpose of the timing marks for each word. With the use of the edit function of the program AUDIOMOD, the block containing the timing mark was found. This was easily accomplished because the timing marks had higher peak values than the actual speech, and AUDIOMOD produced a summary listing of peak values in each block.

Once the timing mark blocks were found, the program BLOCKOUT.FR was used to print out a listing of all 256 sample values in that block. A complete listing of BLOCKOUT.FR can be found in the appendix of this report. From the 256 sample values, the position of the timing marks was found for each word in the pair. It was found that the timing mark consisted of a peak negative voltage of approximately -3.00 volts followed 33 samples later by a peak positive voltage of +3.00 volts.

Once the location of the timing marks was known, the words could be aligned. The number of samples between the distorted word timing marks and the undistorted word timing marks was found from the BLOCKOUT outputs. Then, a program called SHIFT.FR was used to shift the distorted word samples up or down the number of places as needed to align the distorted word file with the undistorted word file. A listing of SHIFT.FR can also be found in the appendix. This program was designed to shift a speech file the number of samples specified in the selected direction. It also zero-fills as necessary the end block from which the last values are shifted and throws away the values shifted out of the first block of the shift. Thus, when analysis is done, the first and last blocks should not be used if the shift was large. Once the shift was completed, a third file was created that contained the shifted version of the distorted word. At this point, 150 word files existed, 50 undistorted word files, 50 distorted but not shifted word flies, and 50 distorted and shifted word flies. All these files were stored on magnetic tape and were loaded into memory only for analysis; they were deleted from memory when analysis was completed.

Computer Implementation of LPC

Once all the words were aligned and stored, the analysis could be started. Since Linear Prediction was selected as the model to be used, two values had to be selected. These inputs to the subroutine AUTO were N, the number of data samples per analysis window, and M, the order of the LPC filter model to be used.

The choice of the analysis interval N is determined by the assumption that the vocal tract movement was negligible on the order of 15-20 ms for most vowels. "Absolute placement of a 15-20 ms interval will not substantially affect the results of either the covariance or the autocorrelation method in most instances" (Ref 12:156). For the autocorrelation method, this meant that pitch asynchronous analysis (the arbitrary placement of the time interval) could be used. Since each sample was 0.125 ms, we could use 128 samples per window for a 16.0 ms window length. Thus, N=128 was used for this analysis. This meant that each block of data could be divided in half and two sets of LPC parameters would be generated for each block.

The second input to AUTO is M, the order of the LPC filter model. It is desirable to use the lowest order possible because the larger the order of the filter, the more coefficients that will be calculated and the longer the program will take. M is limited to a maximum of 21 by the AUTO subroutine due to limited array size statements. M is also limited to a minimum value due to the vocal tract length. Markel and Gray have stated this relationship to be:

$$M = \frac{2Lfs}{c} \tag{3.1}$$

where L is the vocal tract length (previously assumed to be 17 cm), fs is the sample rate of 8000 samples per second, and c is the speed of sound, 34 cm/ms (Ref 12:154). Using these values, M was found to be 8. Thus, M=8 was used for this analysis, and the predictor coefficient vector and the reflection coefficient vector each contained eight elements.

The Fortran IV program AUTOLPC.FR was written to control the use of the subroutine AUTO. The listing of

AUTOLPC.FR can be found in the appendix. When this program is called, it first asks for the filename of the undistorted speech file. Then, it asks for the distorted shifted speech file. Next, the values for N and M are entered. The last two entries are for the first and last blocks to be included in the analysis. This option allows for future analysis of seperate parts of the words. In this analysis, the start block used was block zero and the end block used was the last block before the block containing the first end timing mark.

Once all entries were made in AUTOLPC.FR, the analysis started. Starting with the undistorted word file, the first block was analyzed by subroutine AUTO and the predictor coefficients, reflection coefficients, and total squared error were returned to the main program. Then the next block was analyzed by AUTO, and so forth until the last block designated was analyzed. Next, the distorted and shifted speech file was analyzed in the same manner. The time required to analyze both words averaged two minutes. After both words were analyzed, the main program used the values returned by AUTO and calculated an intelligibility score for that particular word pair. This score was printed to the terminal screen to be recorded by the operator. The method used to calculate the intelligibility score will be discussed in the next section.

After all 50 word pairs were analyzed and scores obtained for each word pair, the 50 scores were averaged to find the mean intelligibility score and the standard deviation associated with the 50 scores about the mean. These values will be presented in the results section of Chapter IV of this report.

Calculation of Objective Intelligiblilty Score

The subroutine AUTO contains three variable names associated with the LPC analysis of Chapter II. The variable A is the array of predictor coefficients $A_i^{(p)}$, the variable RC is the array of reflection coefficients $K^{(p)}$, and ALPHA is the minimum total squared error E_{p+1}. The third variable, ALPHA, is what was used to determine the intelligibility. Markel and Gray (Ref 12:217) state that this minimum total squared error can be considered as residual energy between the actual signal s(n) and the predicted signal generated by the linear prediction process. Each time AUTO is called and run, it returns one value of ALPHA for each 128 samples used. These ALPHA values are summed by the main program for each block of data until the entire word is analyzed. The sum of these ALPHA values is called ALSUMI for the undistorted word file and ALSUMD for the distorted shifted word file. They can be shown by a comparison to the E_{p+1} values from Chapter II as:

$$\text{ALSUMI} = \sum_{\text{all blocks}} \text{ALPHAI} \tag{3.2}$$

$$\text{ALSUMD} = \sum_{\text{all blocks}} \text{ALPHAD} \tag{3.3}$$

where the ALPHAI are the individual E_{p+1} values for the undistorted word file and the ALPHAD are the individual E_{p+1} values for the distorted word file.

Once these two sums were found, the score could be calculated. Two combinations of these values were used to score intelligibility. These two methods were called SUMA and SUMB and are defined as:

$$SUMA = \left[100.0 - \left(\frac{ALSUMI-ALSUMD}{ALSUMI}\right)(100.0)\right]\% \qquad (3.4)$$

$$SUMB = \left[100.0 - \left(\frac{ALSUMD-ALSUMI}{ALSUMD}\right)(100.0)\right]\% \qquad (3.5)$$

Both methods could be used to find an intelligibility score, but SUMB produced values in the 0-100% range that could be used directly as an intelligibility score.

Equation 3.5 can be rewritten as

$$\%Intelligibility = \left[100.0-\left[(\sum E_d - \sum E_u)/E_d\right]/(100.0)\right] \quad (3.6)$$

where the E_u values are the E_{p+1} values of Equation 2.22 for the undistorted word and the E_d values are the E_{p+1} values of the distorted word. Further simplification of 3.6 yields:

$$\%Intelligibility = \left[\sum E_u\right]/\left[\sum E_d\right] \times 100.0\% \qquad (3.7)$$

where the summations are of all the blocks selected for analysis.

This metric (SUMB) was used on all fifty word groups and the average was calculated.

IV. Results

In Chapter III, it was stated that the LPC-10 Vocoder was tested by a Subjective Listener Panel to obtain a subjective evaluation for comparison with the objective measure used in this report. This chapter presents the results of both the subjective testing and the objective testing.

Subjective Testing Results

Subjective Listener Testing was performed on the LPC-10 Vocoder at the facilities of the Biological Acoustics Branch of the Air Force Aerospace Medical Research Laboratory. The Modified Rhyme Test (MRT) was used on a ten member listener panel. The MRT is used by the Biological Acoustics Branch because members of the branch feel it is the best available test that corresponds with Air Force missions. It uses a carrier phrase to simulate actual communications methods and allows six possible responses for each word sent. This has been found to reduce guessing as is possible in the Diagnostic Rhyme Test, which has only two responses per word (Ref 15).

For the tests on the LPC-10 Vocoder, four signal-to-noise levels were used. The level of concern for this report is Level 1, or the "no noise" level. At this level, there is no special background noise being generated and the no-noise characteristics of the vocoder are being tested.

For the Level 1 (zero noise) test, five trials were used. In each trial, one of the ten subjects was selected as the talker and the nine remaining subjects were the listeners. The panel was made up of both males and females who were trained talkers/listeners. Each trial used a fifty word list similar to Table I in Chapter I. Table II lists the results of each trial in two forms. The first value is the number of correct responses out of the fifty words spoken and has a range of values from 0 to 50. The second value is the corrected intelligibility score. This score has been corrected for guessing using the correction method of:

$$\text{Correct Score} = (2.4 \times \text{Number correct out of } 50) - 20 \qquad (4.1)$$

This correction method is the standard method used by the Biological Acoustics Branch for all MRT functions conducted by that branch.

The Table II results show that for zero noise, the LPC-10 Vocoder has an average intelligibility score of 84.91% with a standard deviation of 7.47%. This subjective score can be interpreted to mean that under the above stated conditions, the system will be 84.91% intelligible $\pm$ 7.47%.

Three other noise levels were also tested and are presented in Figure 3. Level 2 is the 95 db noise level, Level 3 is the 105 db noise level, and Level 4 is the 115 db noise level.

TABLE II

Subjective Listener Testing Results

for Modified Rhyme Test
0 db General Noise Case
Old Microphone/5P Mask

Subject	Trial 1	Trial 2	Trial 3	Trial 4	Trial 5
1	40/76.00	40/76.00	39/73.60	TALKER	38/71.20
2	47/92.80	45/88.00	44/85.60	46/90.40	46/90.40
3	TALKER	45/88.00	44/85.60	41/78.40	49/97.60
4	46/90.40	45/88.00	TALKER	43/83.20	48/95.20
5	45/88.00	45/88.00	40/76.00	42/80.80	46/90.40
6	48/95.20	TALKER	46/90.40	47/92.80	47/92.80
7	50/100.0	44/85.60	40/76.00	42/80.80	TALKER
8	40/76.00	40/76.00	36/66.40	35/64.00	43/83.20
9	47/92.80	45/88.00	41/78.40	44/85.60	48/95.20
10	48/95.20	44/85.60	41/78.40	40/76.00	47/92.80
AVG	45.67/89.60	43.67/84.80	41.22/78.93	42.22/81.33	45.78/89.87

TOTAL AVERAGE = 84.91% INTELLIGIBLE
STANDARD DEVIATION = 7.47

notes:

1. All scores are presented as:

   (Number correct out of 50/Corrected Intelligibility Score)

2. All scores are corected for guessing using the formula:

   Corrected Score = (2.4 X Number correct out of 50) - 20

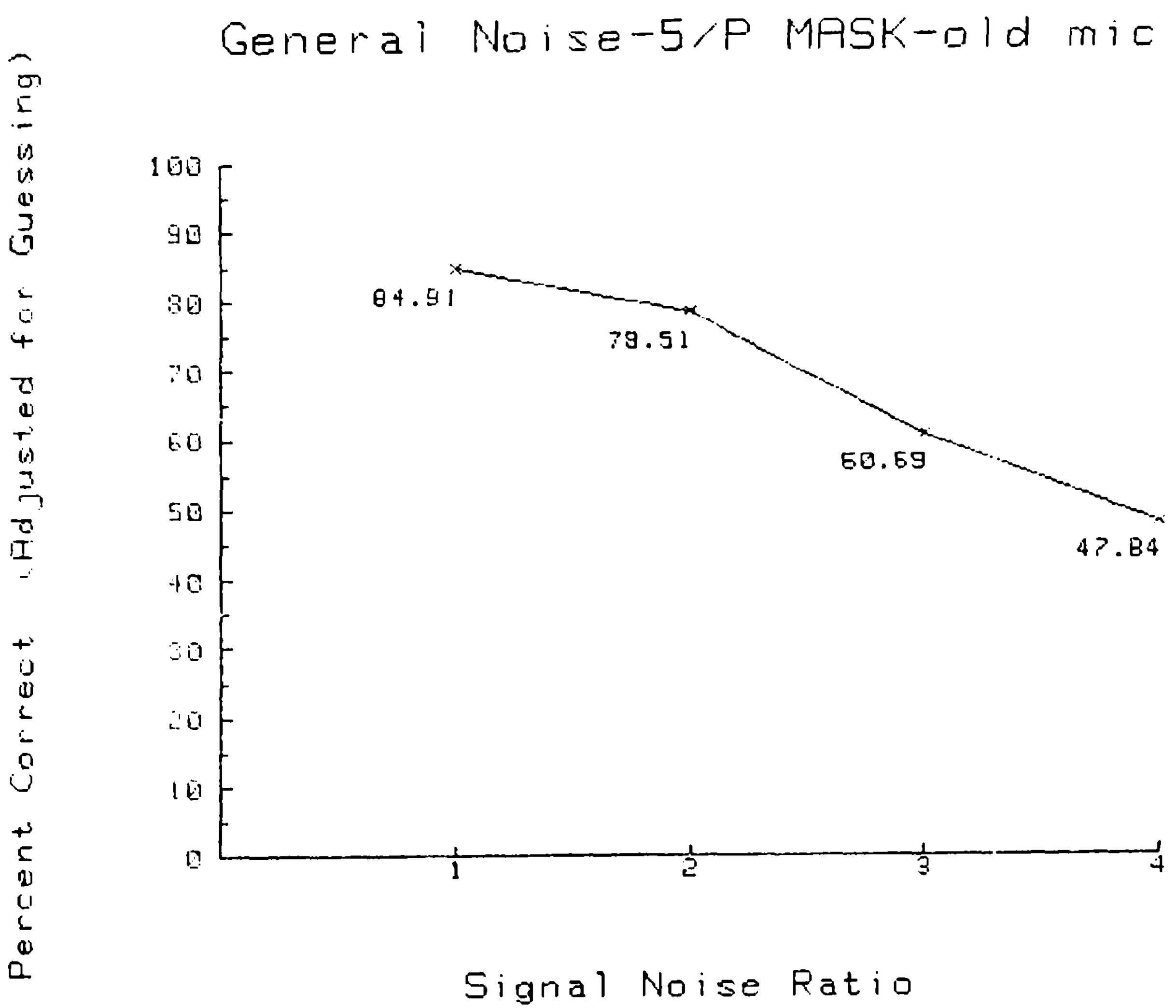

Figure 3. Subjective Testing RESULTS PLOT

Objective Testing Results

Following the procedures outlined in Chapter III of this report, a total of 50 words were analyzed using the LPC measure described. This measure is the total squared error measure of Equation 3.6 and is called SUMB in these results.

Table III presents a word by word listing of results for the calculation of SUMB. In this table, the column labeled Block Length represents the actual number of blocks that were analyzed. For instance, the word TOOK was analyzed from block 0 to block 51 for a total of 52 blocks. The end block was determined by selecting the block containing the end timing mark used for alignment of words. The shorter block lengths were used so that less calculation time was needed. Use of the end block containing the timing mark assured that all of the word was analyzed because the timing mark came after the word was spoken. From Table III, the average value of SUMB was calculated for the fifty words. The average SUMB = 82.99% with a standard deviation of 14.41%. If SUMB is used as a score of intelligibility, it does correspond to the subjective score of 84.91% ± 7.47%. This shows the desired correspondence between the subjective and objective measures used in this report.

Although the objective score is well within one standard deviation of the subjective score, it should be noted that the standard deviation of the objective score is quite large. Because of this large deviation, it would appear that the objective measure could very easily have fallen

TABLE III

Objective Intelligibility Results

WORD FILE	BLOCK LENGTH	SUMB
TOOK	52	84.43
GUST	65	86.57
GANG	66	65.82
PEACH	63	93.62
SUP	68	80.53
BASS	48	71.58
PACK	66	69.69
PIN	60	90.92
COIL	51	85.08
SAD	51	69.92
DUG	61	76.05
TIP	64	114.14
CUFF	66	96.06
GALE	66	70.44
DAY	63	69.32
LAW	62	134.66
TEST	56	97.79
LAY	60	66.10
FEAT	66	62.94
BENT	57	68.51
BIG	52	70.10
SUN	42	90.11
HOT	52	65.79
FIT	45	119.72
TEASE	52	70.32

WORD FILE	BLOCK LENGTH	SUMB
TACK	46	61.74
MAT	44	81.39
FIB	53	88.82
SHOP	64	80.26
WILL	65	85.20
SANE	63	78.72
PANE	55	94.32
FEEL	59	78.15
RED	57	83.50
KILL	59	92.71
DIM	61	97.50
SAME	57	80.53
PEN	63	80.54
CAVE	61	74.56
SIN	61	93.18
PARK	60	70.80
PICK	51	102.43
DIN	44	80.65
BUCK	63	87.96
FOLD	61	80.80
PUN	59	87.19
RAKE	64	74.94
BEAK	62	78.50
SEED	50	78.63
HEAVE	65	86.29

AVERAGE SUMB = (4149.52/50)
= 82.99% INTELLIGIBLE

STD DEV = 14.41

outside one standard deviation of the subjective score. However, this is misleading. If the standard deviation is calculated on just the first 25 words, an average SUMB of 82.81% is obtained but the standard deviation is 18.61%. It can be seen that the more words that were analyzed the smaller the standard deviation became while the average value of SUMB remains nearly the same. Thus, if a larger number than 50, say 200, words were analyzed, it appears that the standard deviation might be reduced to an even more acceptable level under 10%.

V. Conclusions and Recommendations

Conclusions

The major conclusion that may be drawn from this report is that the linear predictive coding total squared error metric examined in Chapters III and IV does perform as an objective intelligibility measure for the no-noise case for the LPC-10 Vocoder under test. It resulted in an intelligibility score of 82.99%, which is within one standard deviation of the subjective score of 84.91%. However, this is a very limited result and should be tested for other systems and noise levels for a better examination of the performance of this metric.

In a report by Ottinger in 1978, a metric was developed that is nearly the same as the metric used in this report. This was Distance Measure 1 in the Ottinger report (Ref 13:25). However, Ottinger found no correlation between his Distance Measure 1 and subjective scores for his system under test. The major difference between the Ottinger method and the method used in this report is that Ottinger did not align the words before analysis. This is the only major difference and has resulted in two totally different results. It appears that for this intelligibility metric to perform properly, the undistorted and distorted words must be aligned before the intelligibility score is calculated. Since the scoring involves the direct comparison between the two words, it is logical that the two words must be aligned for an accurate comparison. This alignment was within one sample out of 128 samples per analysis window for this study.

A third conclusion is that the data base should be large for an accurate objective score. This must be balanced with the goal of reducing the time required for scoring. Each word pair required approximately one-half hour for computer loading, alignment, and analysis. This is not a real-time analysis method, but is still less time demanding than subjective testing. This is because subjective testing usually requires a group of ten or more people for a period of one-to-two days for one complete test run. A major problem in subjective testing is that the listener panel results tend to decay if too many tests are run in one day. The LPC metric is fully computerized and can be continually run with no degradation as long as the computer functions properly.

Recommendations

It is recommended that this study be continued and more research done on various other systems and noise levels. Since this study was done on one communications system with only one noise level, it has very limited results. Further testing should be done to prove or disprove this metric for a wider range of communications systems and noise levels.

It is further recommended that 100 word pairs be used instead of 50 for a more accurate result. This should further reduce the problem of the large standard deviation encountered. If many more than 100 are used, there will be little time savings over subjective listener testing. Therefore, 100 appears to be a good compromise.

A final recommendation is that the Ottinger thesis (Ref 13) be repeated but that the words should be aligned using the method in this report. If this is done, it is possible that one of the other metrics examined in that report may prove to be an even better intelligibility measure.

Bibliography

1. Acoustical Society of America. American National Standard: Methods for the Calculation of the Articulation Index. ANSI S3.5-1969. New York: American National Standards Institute, Inc., 1970.

2. Chambers, A. A Review of Tests for the Evaluation of Speech Communication with Particular Reference to High Speed Low Level Strike Aircraft. DTIC Report, AD-913-601. Royal Aircraft Establishment, England, May 1973.

3. Finkes, P. F., Jr. G-Stress Effects on Speech Recognition (tentative title). Unpublished MS thesis. School of Engineering, Air Force Institute of Technology, Wright-Patterson AFB, Ohio, December 1981.

4. Flanagan, James. Speech Analysis, Sythesis, and Perception. New York: Academic Press Inc., 1965.

5. Gamauf, K. J., and W. J. Hartman. Objective Mesurement of Voice Channel Intelligibility. Final Report, FAA-D-77-153. Institute for Telecommunications Sciences, Department of Commerce, Boulder, Colorado, October 1977.

6. Hartman, W. J., and S. F. Boll. Voice Channel Objective Evaluation Using Linear Predictive Coding. Final Report, FAA-RD-75-189. Institute For Telecommunications Sciences, Department of Commerce, Boulder, Colorado, August 1976.

7. Hoffsletter, E. M., J. Tierney, and O. C. Wheeler. Microprocessor Realization of a Linear Predictive Vocoder. Technical Note 1976-37. Lincoln Laboratory, Massachusetts Institute of Technology, Cambridge, Massachusetts, September 1976.

8. Hubbard, R. W., et al. Objective Measurement Techniques For Evaluating Voice Communications Channels. Final Report, FAA-RD-74-77. Institute for Telecommunications Sciences, Department of Commerce, Boulder, Colorado, July 1974.

9. "IEEE Recommended Practice for Speech Quality Measurements," IEEE Transactions on Audio and Electroacoustics, AU-17 (3):225-246 (September 1969).

10. Jurenko, D. J. Automatic Intelligibility Test Equipment. Final Report, RADC-TR-74-51. Rome Air Development Center, Massachusetts, March 1974.

11. Makhoul, J. "Linear Prediction: A Tutorial Review," Proceedings of the IEEE, 63: 561-580 (April 1975).

12. Markel, John D., and A. H. Gray. Linear Prediction of Speech. New York: Springer-Verlag, 1976.

13. Ottinger, D. M. Objective Measure of Speech Intelligibility Using Linear Predictive Coding. MS Thesis GE/EE/78-35. School of Engineering, Air Force Institute of Technology, Wright-Patterson AFB, Ohio, December 1978.

14. Rabiner, L. R., and R. W. Schafer. Digital Processing of Speech Signals. New Jersey: Prentice-Hall, Inc., 1978.

15. Urban, W. G. Systems Scoring Development: Learning Effect in Modified Rhyme Intelligibility Test. Bell Report No. A70009-337. Arizona Operations, Bell Aerospace Corporation, Fort Huachuca, Arizona, August, 1968.

16. Voiers, W. D. Methods of Predicting User Acceptance of Voice Communication Systems. Final Report, AD-A048-634. Dynastat, Inc., Austin, Texa, June 1976.

APPENDIX

COMPUTER PROGRAM LISTINGS

```
C
C******************************************************************
C******************************************************************
C
C SUBROUTINE:  AUTO
C A SUBROUTINE FOR IMPLEMENTING THE AUTOCORRELATION
C METHOD OF LINEAR PREDICTION ANALYSIS
C
C******************************************************************
C******************************************************************
C
      SUBROUTINE AUTO(N, X, M, A, ALPHA, RC)
C
C   INPUTS:    N - NO. OF DATA POINTS
C              X(N) - INPUT DATA SEQUENCE
C              M - ORDER OF FILTER (M<21, SEE NOTE*)
C  OUTPUTS:    A - FILTER COEFFICIENTS
C              ALPHA - RESIDUAL "ENERGY"
C              RC - REFLECTION COEFFICIENTS
C
C *PROGRAM LIMITED TO M<21 BECAUSE OF DIMENSIONS OF R(.)
C
      DIMENSION X(256), A(260), RC(260)
      DIMENSION R(21)
      MP = M + 1
      DO 20 K=1,MP
        R(K) = 0.
        NK = N - K + 1
        DO 10 NP=1,NK
          N1 = NP + K - 1
          R(K) = R(K) + X(NP)*X(N1)
   10   CONTINUE
   20 CONTINUE
      RC(1) = -R(2)/R(1)
      A(1) = 1.
      A(2) = RC(1)
      ALPHA = R(1) + R(2)*RC(1)
      DO 50 MINC=2,M
        S = 0.
        DO 30 IP=1,MINC
          N1 = MINC - IP + 2
          S = S + R(N1)*A(IP)
   30   CONTINUE
        RC(MINC) = -S/ALPHA
        MH = MINC/2 + 1
        DO 40 IP=2,MH
          IB = MINC - IP + 2
          AT = A(IP) + RC(MINC)*A(IB)
          A(IB) = A(1B) + RC(MINC)*A(IP)
          A(IP) = AT
   40   CONTINUE
          A(MINC+1) = RC(MINC)
```

```
      ALPHA = ALPHA + RC(MINC)*S
      IF (ALPHA) 70, 70, 50
   50 CONTINUE
   60 RETURN
   70 CONTINUE
C
C WARNING - SINGULAR MATRIX
C
      IOUTD = 10
      IOUTP = 1
      WRITE (IOUTD,9999)
      WRITE (IOUTP,9999)
 9999 FORMAT (33H WARNING - SINGULAR MATRIX - AUTO)
      GO TO 60
      END
C
```

```
C*************************************************************
C*************************************************************
C
C                PROGRAM BLOCKOUT.FR
C
C                FORTRAN IV LISTING
C
C                LTJG J.A. KAYSER, USCG
C
C*************************************************************
C*************************************************************
C
C       IFILE-          FILE TO BE PRINTED
C
C       JFILE-          FILENAME OF OUTPUT FILE
C
C       ISTORE-         256 VALUE INTEGER ARRAY USED TO
C                       STORE THE VALUES OF EACH BLOCK
C                       TO BE PRINTED.
C
C       ASTORE-         256 VALUE REAL ARRAY USED TO
C                       STORE THE CONVERTED VOLTAGES
C                       FROM ISTORE
C
C       SBLK-           STARTING BLOCK TO BE PRINTED
C
C       CBLK-           THE TOTAL NUMBER OF BLOCKS
C                       TO BE PRINTED
C
C       EBLK-           LAST BLOCK TO BE PRINTED
C
C       ST-             STATUS CHECK WORD USED IN THE
C                       CALL STAT COMMAND
C
C*************************************************************
C
C       THIS PROGRAM IS USED TO PRINT OUT ANY NUMBER OF
C       BLOCKS IN A GIVEN SPEECH FILE. THE OUTPUT IS
C       CONVERTED TO A REAL VOLTAGE VALUE BETWEEN -5.00
C       VOLTS AND +5.00 VOLTS.
C
C*************************************************************
C
        INTEGER IFILE(13),SBLK,CBLK,ISTORE(256),ST(22),
     :  IBLOCKS,JFILE(13),EBLK
C
        REAL ASTORE(256)
C
C*************************************************************
C
C       ENTER THE INPUT FILENAME TO BE PRINTED OUT AND
C       THE OUTPUT FILENAME DESIRED TO STORE RESULTS
```

```
C
C*************************************************************
C
      TYPE " INPUT FILENAME TO BE PRINTED: "
      READ(11,20) IFILE(1)
 20   FORMAT(S13)
C
      CALL STAT(IFILE,ST,IER)
      IF(IER.NE.1) GO TO 900
      IBLOCKS=ST(9)+1
      TYPE " BLOCK COUNT IS ",IBLOCKS
      CALL OPEN(5,IFILE,2,IER)
      IF(IER.NE.1) GO TO 900
C
      TYPE " INPUT DESIRED OUTPUT FILENAME: "
      READ(11,30) JFILE(1)
 30   FORMAT(S13)
      CALL OPEN(6,JFILE,2,IER)
      IF(IER.NE.1) GO TO 900
C
C*************************************************************
C
C     ENTER STARTING BLOCK AND NUMBER OF BLOCKS TO BE
C     PRINTED. IF THIS EXCEEDS THE LAST BLOCK THEN THE
C     NUMBER OF BLOCKS TO BE PRINTED IS READJUSTED TO
C     STOP AT THE LAST BLOCK.
C
C*************************************************************
C
      ACCEPT " INPUT STARTING BLOCK: ",SBLK
C
      ACCEPT " INPUT NUMBER OF BLOCKS TO BE PRINTED: ",CBLK
      EBLK=SBLK+CBLK
      IF(EBLK.GT.ST(9))TYPE " ADJUSTED END BLOCK TO: ",ST(9)
      IF(EBLK.GT.ST(9))EBLK=ST(9)
 55   CALL RDBLK(5,SBLK,ISTORE,1,IER)
      IF(IER.NE.1) GO TO 900
C
C*************************************************************
C
C     CONVERT EACH BLOCK TO BE PRINTED INTO VOLTAGES AND
C     STORE IN THE ARRAY ASTORE. WRITE ASTORE INTO THE
C     FILE NAMED BY JFILE
C
C*************************************************************
C
      DO 60 I=1,256
         ASTORE(I)=(ISTORE(I)/2048.0)*5.0
 60   CONTINUE
C
      WRITE(6,70)SBLK,IFILE(1)
 70   FORMAT(" "//"  BLOCK NUMBER=",I3,"  FILENAME: ",S13,/)
```

```
      WRITE(6,80)(ASTORE(K),K=1,256)
   80 FORMAT(" ",16F6.2)
C
      SBLK=SBLK+1
      IF(SBLK.LE.EBLK) GO TO 55
C
      TYPE " END OF PROGRAM. YOUR OUTPUT IS "
      WRITE(10,85)JFILE(1)
   85 FORMAT(" LOCATED IN A FILE NAMED:  ",S13)
      GO TO 915
C
C*************************************************************
C
C       ERROR MESSAGE ROUTINE. IT IS USED WHEN THE IER
C       VARIABLE IS RETURNED AS A NON-ONE VALUE DURING
C       ANY CALL COMMAND IN THE PROGRAM.
C
C*************************************************************
C
  900 TYPE " <7><7>** FORTRAN IV SYSTEM ERROR **<7><7> "
      TYPE " ERROR CODE=",IER
      TYPE " PROGRAM ABORTED "
C
C*************************************************************
C
C       END OF PROGRAM. USES A CALL RESET COMMAND TO
C       RESET ALL CHANNELS OPENED.
C
C*************************************************************
C
  915 TYPE " END OF PROGRAM "
      CALL RESET
      END
```

```
C
C*************************************************************
C*************************************************************
C
C               PROGRAM SHIFT.FR
C
C               FORTRAN IV LISTING
C
C               LTJG J.A. KAYSER, USCG
C
C*************************************************************
C*************************************************************
C
C       IBLOCKS-        NUMBER OF BLOCKS IN FILE
C
C       IFILE-          FILE TO BE SHIFTED
C
C       JFILE-          FILENAME OF SHIFTED OUTPUT FILE
C
C       ISHIFT-         NUMBER OF PLACES TO BE SHIFTED
C
C       ISTORE-         256 VALUE ARRAY USED TO STORE
C                       ONE BLOCK OF UNSHIFTED DATA
C                       FROM IFILE.
C
C       JSTORE-         256 VALUE ARRAY USED TO STORE
C                       ONE BLOCK OF SHIFTED DATA FOR
C                       STORAGE IN JFILE
C
C       MOVES-          DIRECTION OF SHIFT (1=UP/2=DOWN)
C
C       SBLK-           STARTING BLOCK OF THE SHIFT
C
C       ST-             STATUS CHECK WORD USED IN
C                       THE CALL STAT COMMAND
C
C*************************************************************
C
C       THIS PROGRAM IS USED TO ALIGN VOICE FILES BY
C       MOVING EACH WORD IN THE FILE BY A NUMBER OF
C       PLACES AS SELECCTED BY THE USER. IT CAN MOVE
C       EITHER UP OR DOWN. THE END BLOCKS ARE ZERO
C       FILLED AS NECESSARY FOR THE SIZE OF THE SHIFT.
C
C*************************************************************
C
        INTEGER IBLOCKS,ISHIFT,IFILE(13),MOVES,JFILE(13),
     :  ISTORE(256),JSTORE(256),IER,ST(22),SBLK
C
C*************************************************************
C
C       ZERO OUT THE ISTORE AND JSTORE ARRAYS AT START
```

```
C
C*************************************************************
C
      DO 10 I=1,256
      ISTORE(I)=0
      JSTORE(I)=0
 10   CONTINUE
C
C*************************************************************
C
C     ENTER THE INPUT FILENAME TO BE SHIFTED AND THE
C     OUTPUT FILENAME TO STORE RESULTS OF THE SHIFT
C
C*************************************************************
C
      TYPE "  ENTER FILE NAME TO BE SHIFTED: "
      READ (11,15) IFILE(1)
 15   FORMAT(S13)
      CALL STAT(IFILE,ST,IER)
      IF(IER.NE.1) GO TO 900
      IBLOCKS=ST(9)+1
      TYPE "  BLOCK COUNT IS ",IBLOCKS
      CALL OPEN(5,IFILE,2,IER)
      IF(IER.NE.1) GO TO 900
      TYPE "  FILE IS OPEN. "
      TYPE "  ENTER OUTPUT FILENAME DESIRED: "
      READ (11,5) JFILE(1)
 5    FORMAT(S13)
C
C*************************************************************
C
C     INPUT THE AMMOUNT OF THE SHIFT AND THE DIRECTION
C     OF THE SHIFT. 1=UP AND 2=DOWN.
C
C*************************************************************
C
      ACCEPT "<15> HOW MANY PLACES WILL THE SHIFT INVOLVE?<15>
     :SHIFT= ",ISHIFT
      IF(ISHIFT.LE.256) GO TO 20
      TYPE " EDIT THIS FILE. YOUR SHIFT IS GREATER THAN 256."
      GO TO 910
C
 20   TYPE " SHIFT UP OR DOWN. INPUT 1 FOR UP OR 2 FOR DOWN."
      ACCEPT " DIRECTION= ",MOVES
      IF(MOVES.EQ.2) GO TO 500
      IF(MOVES.EQ.1) GO TO 25
      TYPE "  YOU MUST SELECT A 1 OR 2 ONLY!!!<15>"
      GO TO 20
C
C*************************************************************
C
C     THIS IS THE START OF THE SHIFT UP ROUTINE. IT
```

```
C       STARTS WITH THE LAST BLOCK IN IFILE AND SHIFTS
C       UP EACH BLOCK AN AMMOUNT OF ISHIFT UNTIL IT HAS
C       SHIFTED UP THE ENTIRE FIRST BLOCK. IT STORES
C       THE RESULTS IN THE FILE JFILE VIA THE JSTORE ARRAY
C
C*****************************************************************
C
  25    TYPE "  YOU HAVE SELECTED A SHIFT OF "
        TYPE ISHIFT," PLACES IN THE UP DIRECTION."
        CALL OPEN(6,JFILE,2,IER)
        IF(IER.NE.1) GO TO 900
        SBLK=ST(9)
        CALL RDBLK(5,SBLK,ISTORE,1,IER)
        IF(IER.NE.1) GO TO 900
        II=256-ISHIFT
        JJ=256
C
  30    JSTORE(JJ)=ISTORE(II)
        JJ=JJ-1
        II=II-1
        IF(II.GE.1) GO TO 30
C
  35    SBLK=SBLK-1
        CALL RDBLK(5,SBLK,ISTORE,1,IER)
        II=256
C
  40    JSTORE(JJ)=ISTORE(II)
        JJ=JJ-1
        II=II-1
        IF(II.LT.1) GO TO 50
        IF(JJ.GE.1) GO TO 40
C
        CALL WRBLK(6,SBLK+1,JSTORE,1,IER)
        IF(IER.NE.1) GO TO 900
        JJ=256
        DO 45 I=1,256
           JSTORE(I)=0
  45    CONTINUE
        GO TO 40
C
  50    IF(SBLK.GT.0) GO TO 35
        CALL WRBLK(6,SBLK,JSTORE,1,IER)
        IF(IER.NE.1) GO TO 900
        GO TO 915
C
C*****************************************************************
C
C       THIS IS THE START OF THE SHIFT DOWN ROUTINE. IT
C       STARTS WITH THE FIRST BLOCK IN IFILE AND SHIFTS
C       DOWN EACH BLOCK AN AMMOUNT ISHIFT UNTIL IT HAS
C       SHIFTED THE ENTIRE LAST BLOCK. IT STORES THE
C       RESULTS IN THE FILE JFILE VIA THE JSTORE ARRAY.
```

```
C
C***********************************************************************
C
 500    TYPE "  YOU HAVE SELECTED A SHIFT OF "
        TYPE ISHIFT," PLACES IN THE DOWN DIRECTION."
        CALL OPEN(6,JFILE,2,IER)
        IF(IER.NE.1) GO TO 900
        SBLK=0
        CALL RDBLK(5,SBLK,ISTORE,1,IER)
        IF(IER.NE.1) GO TO 900
        II=ISHIFT
        JJ=1
C
 510    JSTORE(JJ)=ISTORE(II)
        JJ=JJ+1
        II=II+1
        IF(II.LE.256) GO TO 510
C
 520    SBLK=SBLK+1
        CALL RDBLK(5,SBLK,ISTORE,1,IER)
        IF(IER.NE.1) GO TO 900
        II=1
C
 530    JSTORE(JJ)=ISTORE(II)
        JJ=JJ+1
        II=II+1
        IF(II.GT.256) GO TO 540
        IF(JJ.LE.256) GO TO 530
C
        CALL WRBLK(6,SBLK-1,JSTORE,1,IER)
        IF(IER.NE.1) GO TO 900
        JJ=1
        DO 535 I=1,256
          JSTORE(I)=0
 535    CONTINUE
        GO TO 530
C
 540    IF(SBLK.LT.IBLOCKS-1) GO TO 520
        CALL WRBLK(6,SBLK,JSTORE,1,IER)
        IF(IER.NE.1) GO TO 900
        GO TO 915
C
C***********************************************************************
C
C       THIS IS THE ERROR MESSAGE ROUTINE. IT IS USED
C       WHEN THE IER VARIABLE IS RETURNED AS A NON-ONE
C       VALUE DURING ANY CALL COMMAND IN THE PROGRAM.
C
C***********************************************************************
C
 900    TYPE " <7><7>** FORTRAN IV SYSTEM ERROR **<7><7>"
        TYPE " ERROR CODE = ",IER
```

```
      TYPE " RETURNED ON A CALL COMMAND."
910   TYPE "  PROGRAM ABORTED <7><7>"
C
C***********************************************************
C
C     END OF PROGRAM. USES A CALL RESET COMMAND
C     TO RESET ALL CHANNELS OPENED.
C
C***********************************************************
C
915   TYPE "  END OF PROGRAM. "
      CALL RESET
      END
```

```
C*************************************************************
C*************************************************************
C
C                 PROGRAM AUTOLPC.FR
C
C                 FORTRAN IV LISTING
C
C                 LTJG JEFFREY A. KAYSER, USCG
C
C*************************************************************
C*************************************************************
C
        INTEGER ISTORE(256),JSTORE(256),IFILE(13),JFILE(13),
     :  AST(22),DST(22),ST,IBLOCKS,M,SBLK,IER,J,K
        REAL ALPHAI,ALPHAD,X(256)
        DIMENSION AI(260),AD(260),RCI(260),RCD(260)
C
C*************************************************************
C
C       THIS SECTION READS IN THE NAMES OF THE FILES
C       TO BE ANALYZED USING THE LPC SUBROUTINE CALLED
C       AUTO.FR. THIS IS A FORTRAN IV SUBROUTINE.
C
C*************************************************************
C
        TYPE " ENTER ANALOG FILE TO BE TESTED: "
        READ(11,10) IFILE(1)
        TYPE " ENTER DISTORTED FILE TO BE TESTED: "
        READ(11,10) JFILE(1)
 10     FORMAT(S13)
        CALL STAT(IFILE,AST,IER)
        IF(IER.NE.1) GO TO 900
        CALL STAT(JFILE,DST,IER)
        IF(IER.NE.1)GO TO 900
        ST=DST(9)
        IF(AST(9).LT.DST(9)) ST=AST(9)
        IBLOCKS=ST+1
        TYPE " MAXIMUM BLOCK COUNT IS ",IBLOCKS
        CALL OPEN(1,IFILE,2,IER)
        IF(IER.NE.1) GO TO 900
        CALL OPEN(2,JFILE,2,IER)
        IF(IER.NE.1) GO TO 900
        CALL OPEN(3,"JAK78",2,IER)
        IF(IER.NE.1) GO TO 900
        ALSUMI=0.00
        ALSUMD=0.00
C
C*************************************************************
C
C       THE FOLLOWING SECTION READS IN THE DESIRED FILTER
C       SIZE, NUMBER OF POINTS PER ANALYSIS WINDOW, AND
C       THE STARTING AND ENDING BLOCKS DESIRED. NOTE THAT
```

```
C       THE FILTER SIZE MUST BE LESS THAN THE NUMBER OF
C       POINTS PER WINDOW AND A MAXIMUM OF 21.
C
C*************************************************************
C
  20    TYPE " ENTER LPC FILTER SIZE (<21): "
        ACCEPT " SIZE= ",M
        IF(M.LT.21) GO TO 30
        TYPE " MUST BE LESS THAN 21 ! "
        GO TO 20
C
  30    ACCEPT " NUMBER OF POINTS= ",IVAL
        ACCEPT " START BLOCK=",SBLK
        ACCEPT " END BLOCK=",EBLK
        IF(EBLK.GT.ST) EBLK=ST
        LOOP=SBLK
C
C*************************************************************
C
C       THE FOLLOWING SECTION READS EACH BLOCK OF THE
C       SELECTED SPEECH FILE AND CALLS THE SUBROUTINE
C       AUTO TO CALCULATE THE LPC COEFFICIENTS.
C
C*************************************************************
C
  35    CALL RDBLK(1,SBLK,ISTORE,1,IER)
        IF(IER.NE.1) GO TO 900
        N=IVAL
        J=0
C
  40    DO 50 I=1,IVAL
          L=I+J
          X(I)=((ISTORE(L)/2048.0)*5.0)
  50    CONTINUE
C
        CALL AUTO(N,X,M,AI,ALPHAI,RCI)
        ALSUMI=ALSUMI+ALPHAI
        REF=0.0
        DO 55 II=1,M
          REF=REF+RCI(II)
  55    CONTINUE
        J=J+IVAL
        IF(J.LT.256) GO TO 40
C
        SBLK=SBLK+1
        IF(SBLK.LE.EBLK) GO TO 35
        SBLK=LOOP
  75    CALL RDBLK(2,SBLK,JSTORE,1,IER)
        IF(IER.NE.1) GO TO 900
        J=0
  90    DO 100 I=1,IVAL
          L=I+J
```

```
        X(I)=((JSTORE(L)/2048.0)*5.0)
 100    CONTINUE
C
        CALL AUTO(N,X,M,AD,ALPHAD,RCD)
        ALSUMD=ALSUMD+ALPHAD
        REF=0.0
        DO 105 II=1,M
          REF=REF+RCD(II)
  105   CONTINUE
        J=J+IVAL
        IF(J.LT.256) GO TO 90
        SBLK=SBLK+1
        IF(SBLK.LE.EBLK)GO TO 75
        GO TO 915
C
C
C*************************************************************
C
C       ERROR ROUTINE SECTION. THIS SECCTION IS USED
C       WHEN A NON-ONE VALUE IS RETURNED BY THE IER
C       VARIABLE ON ANY CALL COMMAND.
C
CC************************************************************
C
 900    TYPE " <7><7> FORTRAN IV SYSTEM ERROR <7><7>"
        TYPE " ERROR =",IER
        TYPE " ERROR IS ON A CALL COMMAND"
C
C*************************************************************
C
C       END OF PROGRAM. THE RESULTS ARE PRINTED TO
C       THE SCREEN AS A RESULT CALLED SUMB.
C
C*************************************************************
C
915     TYPE " END OF PROGRAM"
        SUMB=(100.0-((ALSUMD-ALSUMI)/ALSUMD)*100.0)
        TYPE "  SUMB=",SUMB
        TYPE "  ALSUMI=",ALSUMI,"  ALSUMD=",ALSUMD
        CALL RESET
        END
```

Vita

Jeffrey A. Kaysor was born on 25 September 1956 in Dayton, Ohio. He was graduated from high school in 1974 and received an appointment to the United States Coast Guard Academy. He was graduated from the Coast Guard Academy in 1978 with the degree of Bachelor of Science in Electical Engineering with Honors. He was commissioned an Ensign and served two years aboard the Coast Guard Cutter Chase (WHEC-718) as a student engineer and then assistant engineer and Damage Control Officer. He was then selected for graduate training in Electrical Engineering and assigned to the School of Engineering, Air Force Institute of Technology.

Permanent Address: 209 Bonner Street
Dayton, Ohio 45410

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER
AFIT/GE/EE/81D-30

2. GOVT ACCESSION NO.
AD-A115 515

3. RECIPIENT'S CATALOG NUMBER

4. TITLE (and Subtitle)
THE CORRELATION BETWEEN SUBJECTIVE AND OBJECTIVE MEASURES OF CODED SPEECH QUALITY AND INTELLIGIBILITY FOLLOWING NOISE CORRUPTION

5. TYPE OF REPORT & PERIOD COVERED
MS Thesis

6. PERFORMING ORG. REPORT NUMBER

7. AUTHOR(s)
Jeffrey A. Kayser
Lt.(j.g.) USCG

8. CONTRACT OR GRANT NUMBER(s)

9. PERFORMING ORGANIZATION NAME AND ADDRESS
Air Force Institute of Technology (AFIT-EN)
Wright-Patterson AFB, Ohio 45433

10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS

11. CONTROLLING OFFICE NAME AND ADDRESS
RADC/EEV Maj. Anthony Martinez
Hanscom AFB, Mass. 01731

12. REPORT DATE
December, 1981

13. NUMBER OF PAGES
66

14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office)

15. SECURITY CLASS. (of this report)
Unclassified

15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited

15 APR 1982

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

Dean for Research and Professional Development
Air Force Institute of Technology (ATC)
Wright-Patterson AFB, OH 45433

18. SUPPLEMENTARY NOTES

Approved for public release; IAW AFR 190-17

~~Fredric C. Lynch, Major~~, USAF
~~Director of Public Affairs~~

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Intelligibility Measurements
Linear Predictive Coding
Voice Systems

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

A scoring metric of speech intelligibility based on linear predictive coding (LPC) was developed and evaluated. The data base used for evaluating the metric consisted of a list of 50 words from the Modified Rhyme Test. The list was transmitted over a LPC-10 Vocoder with no background noise. The list was scored subjectively for intelligibility by a trained listener panel. The subjective scores were used to judge the effectiveness

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered) of the subjective metric.

The LPC scoring metric was calculated for the list of words and compared to the subjective scoring. The intelligibility score for the objective scoring metric was 82.99% with a standard deviation of 14.41%. The score for the subjective listener testing was 84.91% with a standard deviation of 7.47%. This shows a possible correlation between the objective LPC scoring metric and standard subjective listener scoring methods.